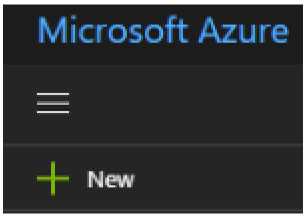
Exercise 3: Deploy Linux RDMA InfiniBand cluster with Azure CLI

Duration: 45 minutes

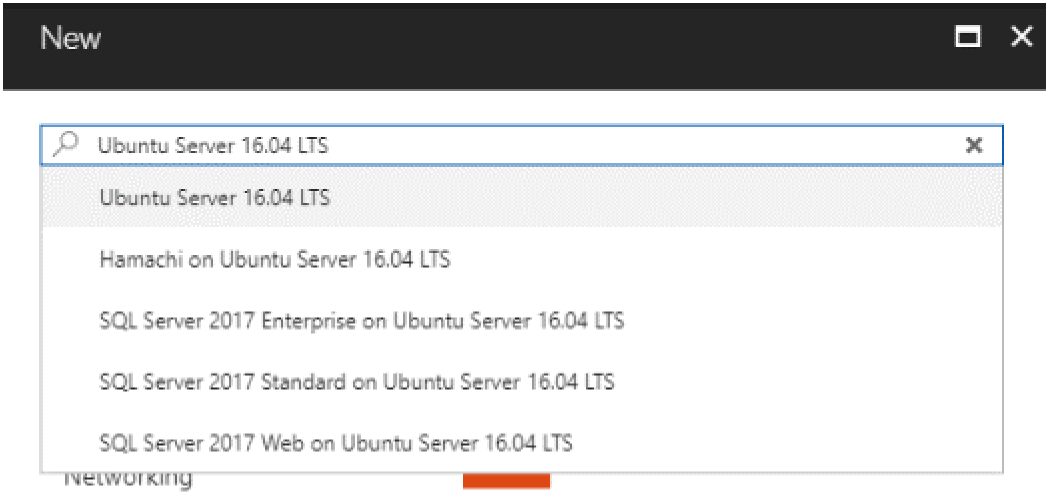
Preparation: Deploy Azure CLI Linux box

In this task you will deploy a Ubuntu 16.04 LTS VM and install Azure CLI in it, as prerequisite for executing further steps of the exercise.

1. Using a browser, navigate to the Azure Portal.
2. Select **+ New** from the menu.

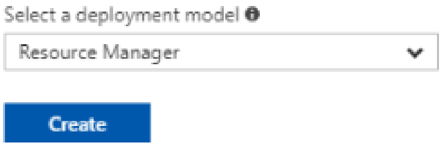


3. In the Search the Marketplace text box, enter “**Ubuntu Server 16.04 LTS**” and select the same in the drop-down list that appears.


New
Ubuntu Server 16.04 LTS
Hamachi on Ubuntu Server 16.04 LTS
SQL Server 2017 Enterprise on Ubuntu Server 16.04 LTS
SQL Server 2017 Standard on Ubuntu Server 16.04 LTS
SQL Server 2017 Web on Ubuntu Server 16.04 LTS
Networking

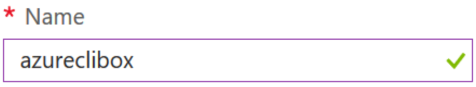
Ubuntu Server 16.04 LTS

4. On the Ubuntu Server 16.04 LTS blade, leave the selection of **Resource Manager** for Select a deployment model, and select **Create**.



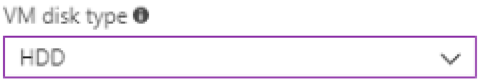
Select a deployment model: Resource Manager

5. On the Basics blade, enter a **Name** for the new VM.



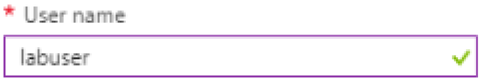
* Name: azureclibox

6. For VM disk type, select **HDD**.



VM disk type: HDD

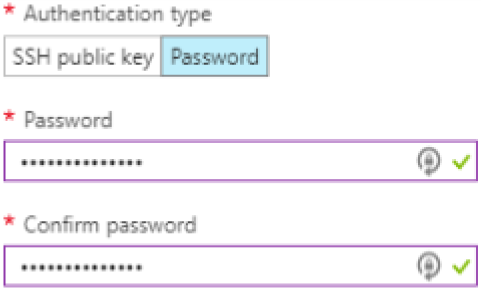
7. For User name, enter a username, e.g. **labuser**.



* User name

labuser ✓

8. For Authentication type, select **Password** and enter a password in the fields that appear. Take note of the user name and password you supply here, as you will use it later to SSH into the VM.



* Authentication type

SSH public key | Password

* Password

•••••••••••••• ✓

* Confirm password

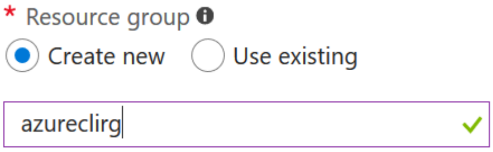
•••••••••••••• ✓

9. Choose your **Azure Subscription**.

Subscription

[redacted]

10. For the Resource Group, select **Create new** and type the name, e.g. **azureclirg**.



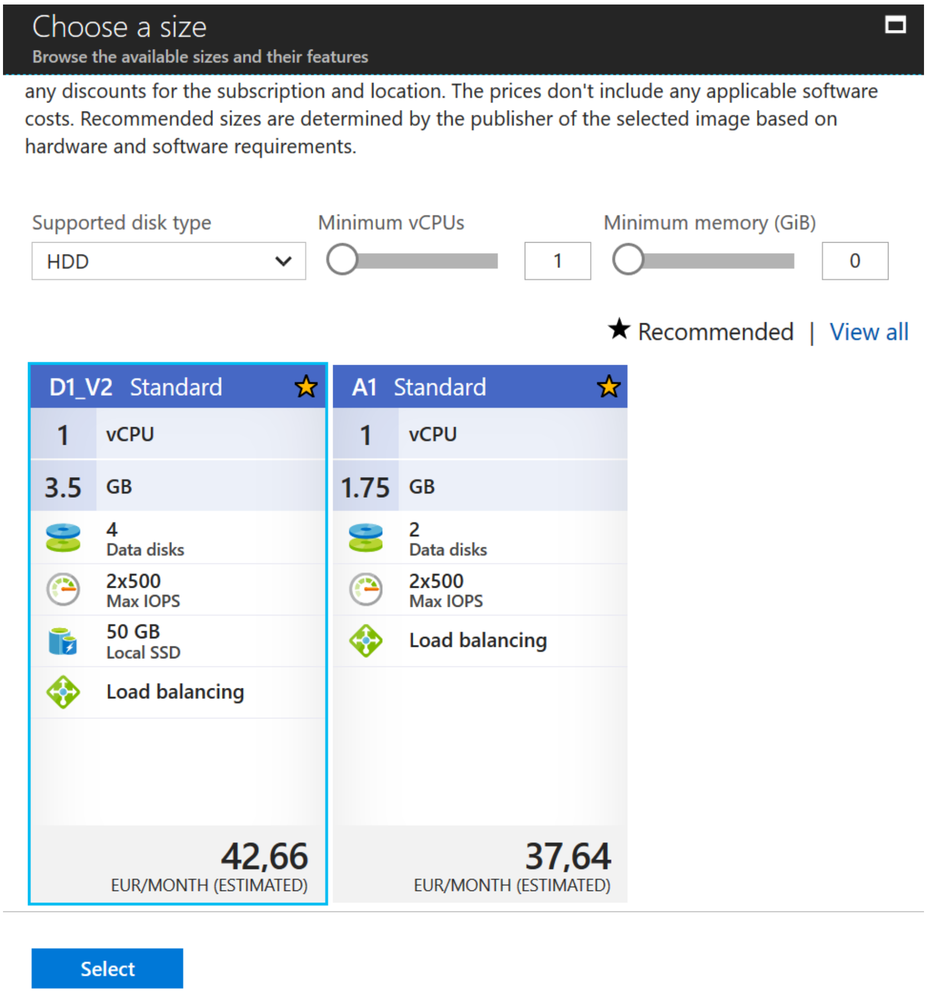
* Resource group

● Create new ○ Use existing

azureclirg ✓

11. Choose a **location,** e.g. **West Europe**.
12. Select **OK**.

13. On the Choose a Size blade, select **D1_V2 Standard** and then click **Select**.



Choose a size

Browse the available sizes and their features

any discounts for the subscription and location. The prices don't include any applicable software costs. Recommended sizes are determined by the publisher of the selected image based on hardware and software requirements.

Supported disk type: HDD

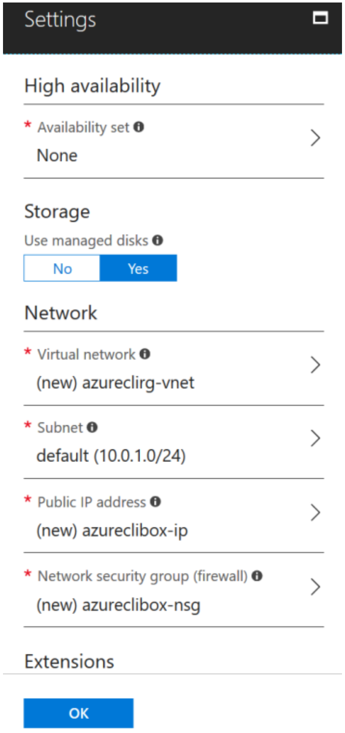
Minimum vCPUs: 1

Minimum memory (GiB): 0

★ Recommended | View all

D1_V2 Standard ★	A1 Standard ★
1 vCPU	1 vCPU
3.5 GB	1.75 GB
4 Data disks	2 Data disks
2x500 Max IOPS	2x500 Max IOPS
50 GB Local SSD	Load balancing
Load balancing	
42,66 EUR/MONTH (ESTIMATED)	37,64 EUR/MONTH (ESTIMATED)

14. On the Settings blade, leave the defaults and select **OK**.



Settings

High availability

* Availability set: None

Storage

Use managed disks: No [✓ Yes]

Network

* Virtual network: (new) azureclirg-vnet

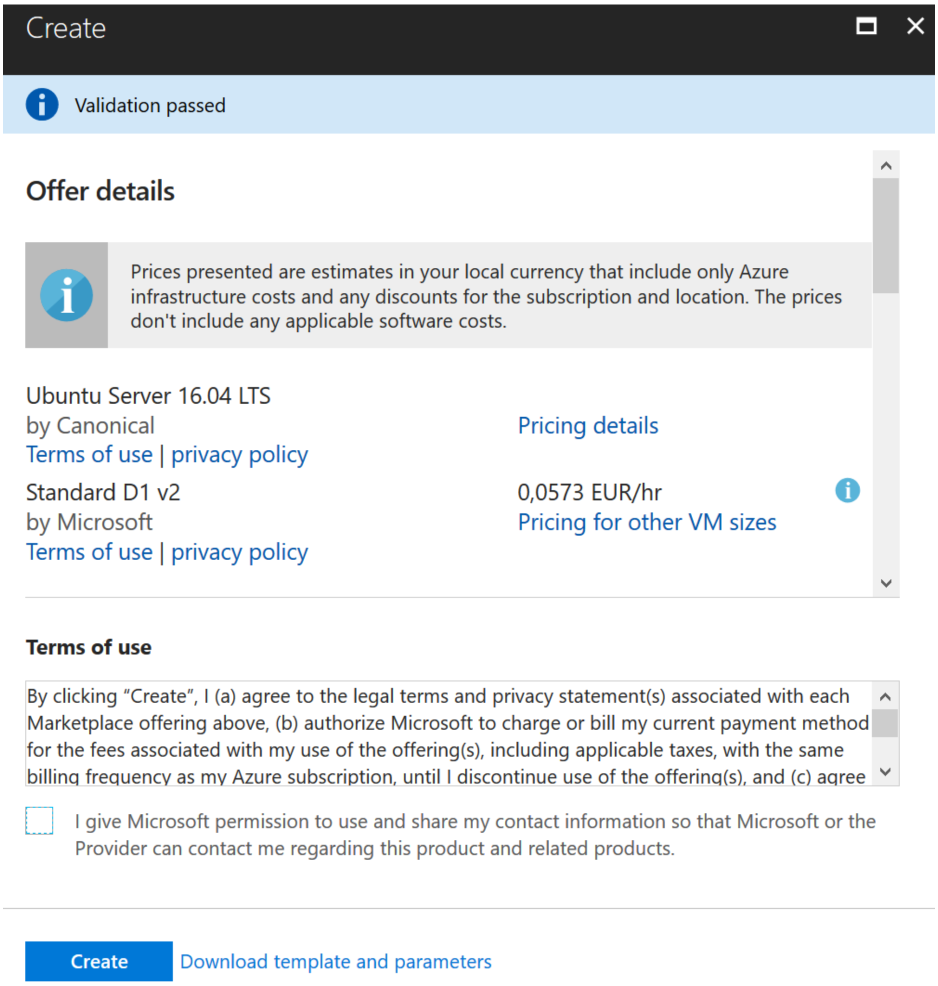
* Subnet: default (10.0.1.0/24)

* Public IP address: (new) azureclibox-ip

* Network security group (firewall): (new) azureclibox-nsg

Extensions

15. On the Create blade, select **Create**.



Create

Validation passed

Offer details

Prices presented are estimates in your local currency that include only Azure infrastructure costs and any discounts for the subscription and location. The prices don't include any applicable software costs.

Ubuntu Server 16.04 LTS
by Canonical
Terms of use | privacy policy

Pricing details

Standard D1 v2
by Microsoft
Terms of use | privacy policy

0,0573 EUR/hr
Pricing for other VM sizes

Terms of use

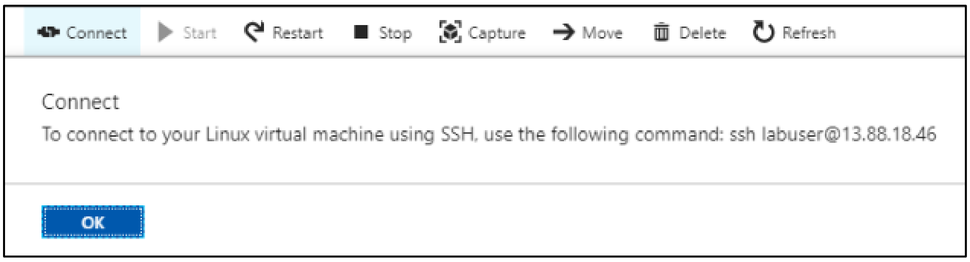
By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with each Marketplace offering above, (b) authorize Microsoft to charge or bill my current payment method for the fees associated with my use of the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s), and (c) agree

□ I give Microsoft permission to use and share my contact information so that Microsoft or the Provider can contact me regarding this product and related products.

It will take about 3-5 minutes to deploy the VM.

16. Once the VM is ready, the blade for the VM will open automatically in the Azure Portal.
17. In the control bar of the VM blade, select **Connect**.

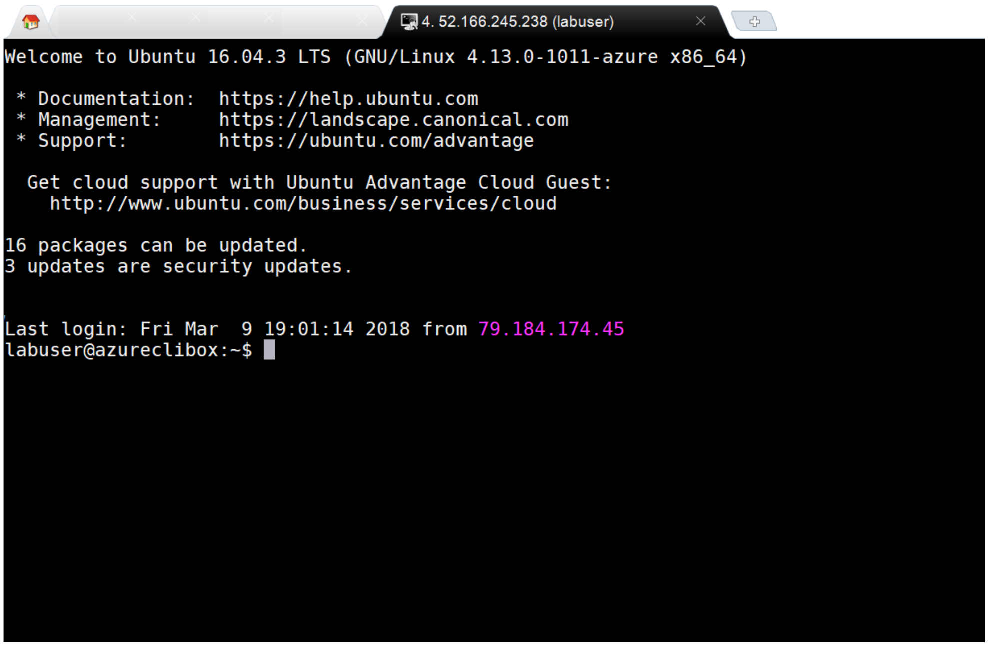
18. A dialog will appear showing the SSH command line to use to connect to the VM. Take note of the command, it includes the username (labuser in the below) and IP address (13.88.18.146 in the below) that will be used to access the VM.



Connect

To connect to your Linux virtual machine using SSH, use the following command: ssh labuser@13.88.18.46

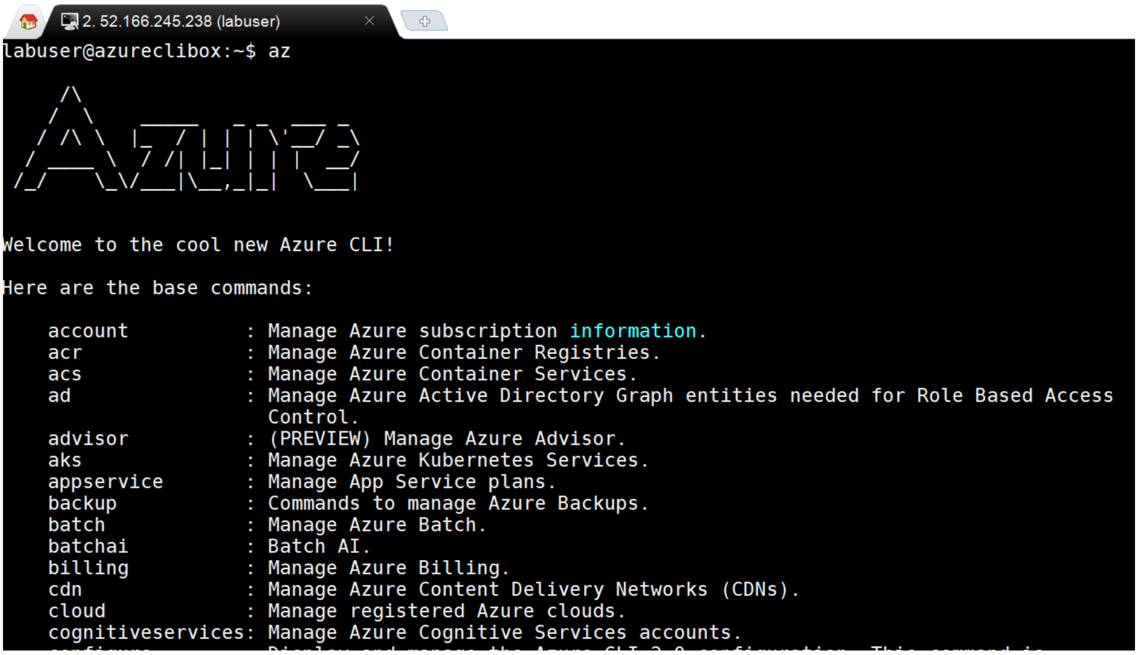
19. Using your favorite SSH client, login into the VM. Be sure to provide the username and password you specified when creating the VM.



```
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.13.0-1011-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

16 packages can be updated.
3 updates are security updates.


Last login: Fri Mar  9 19:01:14 2018 from 79.184.174.45
labuser@azureclibox:~$
```

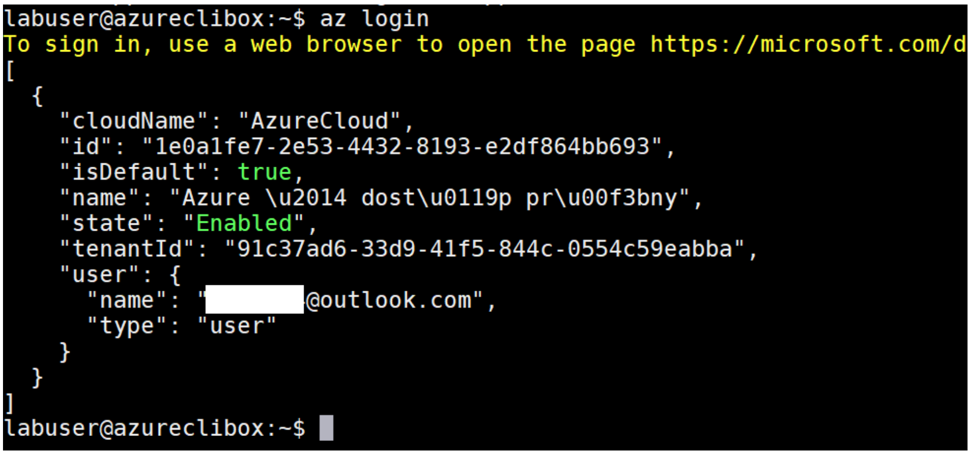
20. Navigate to https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-apt?view=azure-cli-latest and follow the instructions to install Azure CLI in your Ubuntu VM.
21. Verify that the Azure CLI is installed by running:
    `$ az`

    If you see output similar to the following the install was successful.



```
labuser@azureclibox:~$ az

     /\
    /  \    _____   _ _  ___ _
   / /\ \  |_  / | | | \'__/ _\
  / ____ \  / /| |_| | | |  __/
 /_/    \_\/___|\__,_|_|  \___|


Welcome to the cool new Azure CLI!

Here are the base commands:

    account          : Manage Azure subscription information.
    acr              : Manage Azure Container Registries.
    acs              : Manage Azure Container Services.
    ad               : Manage Azure Active Directory Graph entities needed for Role Based Access
                       Control.
    advisor          : (PREVIEW) Manage Azure Advisor.
    aks              : Manage Azure Kubernetes Services.
    appservice       : Manage App Service plans.
    backup           : Commands to manage Azure Backups.
    batch            : Manage Azure Batch.
    batchai          : Batch AI.
    billing          : Manage Azure Billing.
    cdn              : Manage Azure Content Delivery Networks (CDNs).
    cloud            : Manage registered Azure clouds.
    cognitiveservices: Manage Azure Cognitive Services accounts.
```

22. Login to your account with Azure CLI
    `$ az login`
    Follow the provided instruction – go to URL https://microsoft.com/devicelogin in a web-browser and enter the provided code string. Then login to your Microsoft account and close the web-browser page.
    After successful login you should see output similar to the following:

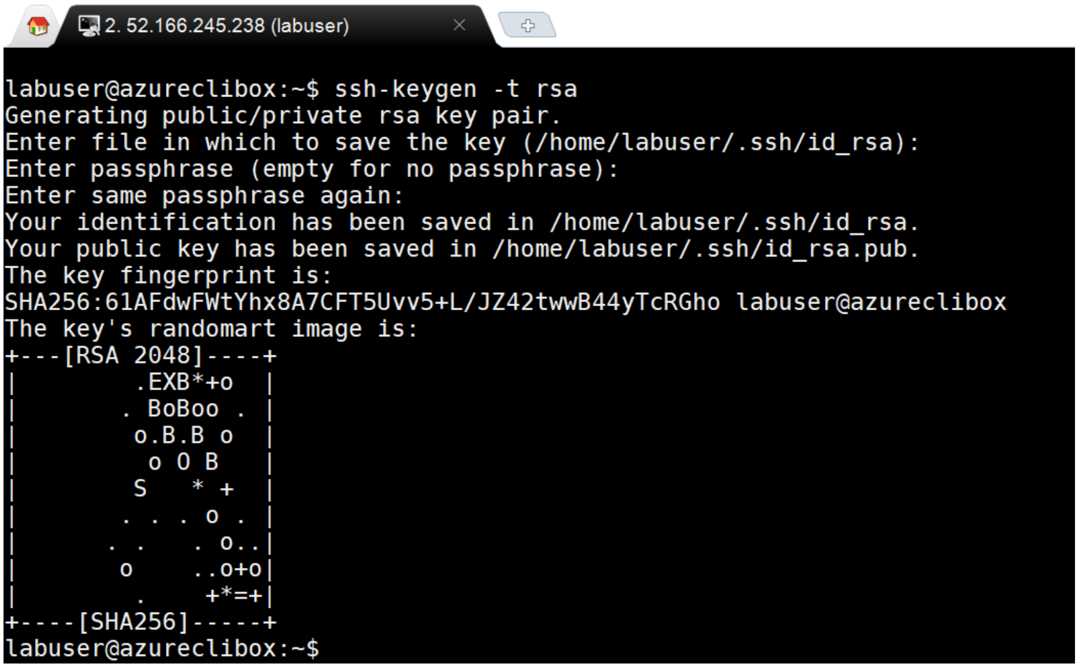


```
labuser@azureclibox:~$ az login
To sign in, use a web browser to open the page https://microsoft.com/d
[
  {
    "cloudName": "AzureCloud",
    "id": "1e0a1fe7-2e53-4432-8193-e2df864bb693",
    "isDefault": true,
    "name": "Azure \u2014 dost\u0119p pr\u00f3bny",
    "state": "Enabled",
    "tenantId": "91c37ad6-33d9-41f5-844c-0554c59eabba",
    "user": {
      "name": "[redacted]@outlook.com",
      "type": "user"
    }
  }
]
labuser@azureclibox:~$ 
```

Now using Azure CLI commands you will deploy a simple CentOS7.1 cluster with RDMA InfiniBand interconnect. The cluster will be deployed using Azure VM scaleset service.

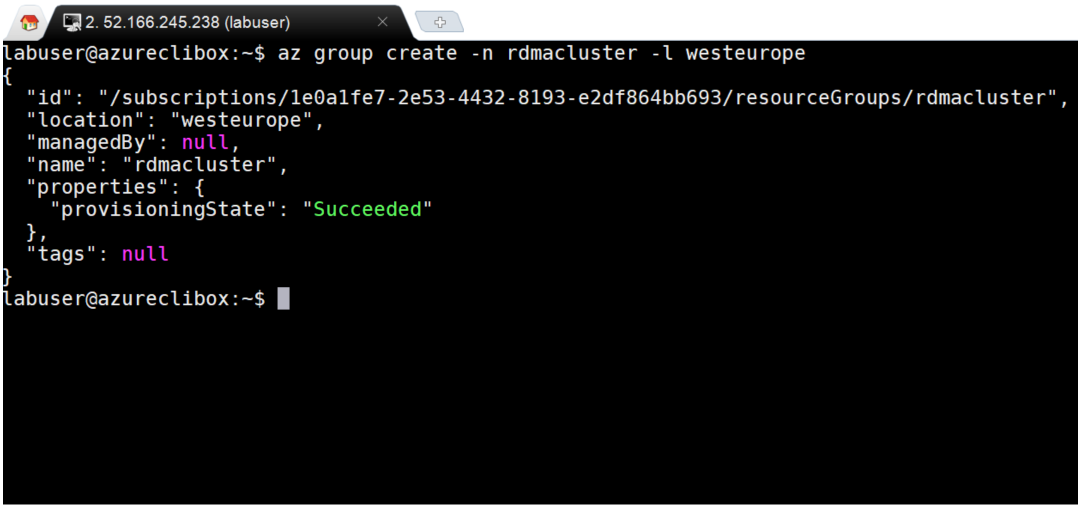
Step 1: Deploy CentOS7.1 cluster with RDMA InfiniBand

1. Open your favourite SSH client and login to Azure CLI Linux box.
2. Generate private/public SSH keys
   `$ ssh-keygen -t rsa`
   Press [ENTER] on all prompts.



```
labuser@azureclibox:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/labuser/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/labuser/.ssh/id_rsa.
Your public key has been saved in /home/labuser/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:61AFdwFWtYhx8A7CFT5Uvv5+L/JZ42twwB44yTcRGho labuser@azureclibox
The key's randomart image is:
+---[RSA 2048]----+
|        .EXB*+o  |
|       . BoBoo . |
|        o.B.B o  |
|         o O B   |
|        S   * +  |
|       . . . o . |
|      . .   . o..|
|       o    ..o+o|
|        .    +*=+|
+----[SHA256]-----+
labuser@azureclibox:~$
```

3. Create resource group for the cluster resources:
   `$ az group create -n rdmacluster -l westeurope`



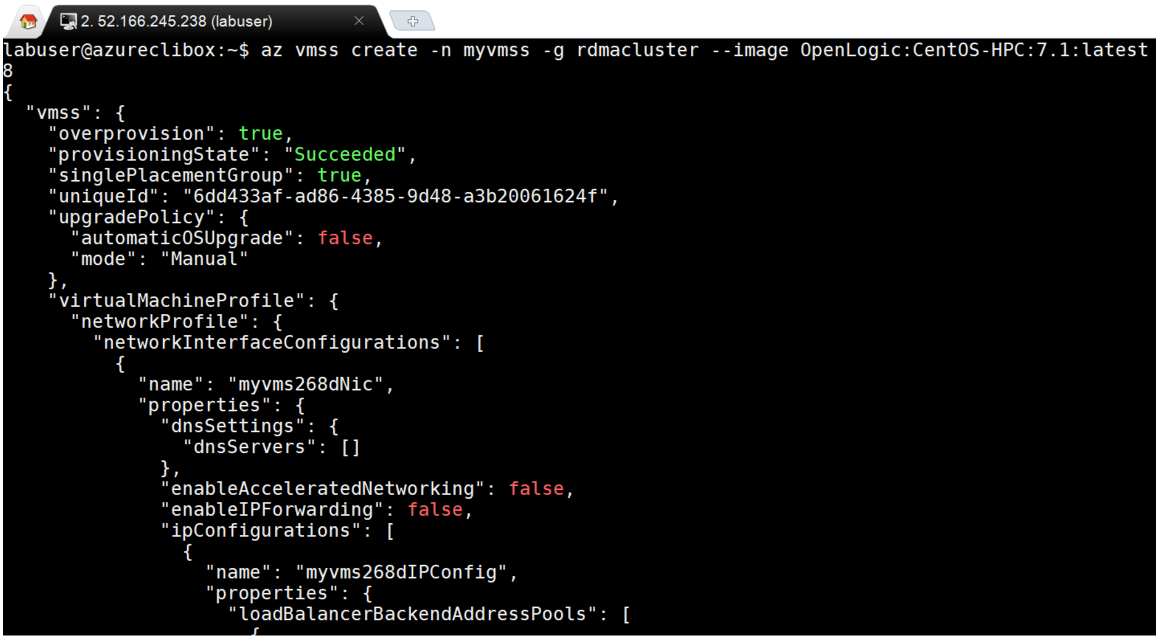
```
labuser@azureclibox:~$ az group create -n rdmacluster -l westeurope
{
  "id": "/subscriptions/1e0a1fe7-2e53-4432-8193-e2df864bb693/resourceGroups/rdmacluster",
  "location": "westeurope",
  "managedBy": null,
  "name": "rdmacluster",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}
labuser@azureclibox:~$
```

Note: The following steps will fail with your Azure trial subscription due to low quota for RDMA enabled VMs. You will need to raise the quota for RDMA InfiniBand enabled VMs to execute the rest of this exercise. Please refer to the instructor on how to continue.

4. Create VM scaleset from Azure Marketplace CentOS7.1-HPC OS image:
   $ az vmss create -n myvmss -g rdmacluster --image OpenLogic:CentOS-HPC:7.1:latest --vm-sku Standard_H16r --instance-count 4

   Here is description of the used command parameters:
   -n myvmss – name of VM scaleset
   -g rdmacluster – resource group for deployment
   --image OpenLogic:CentOS-HPC:7.1:latest – the latest CentOS7.1 OS image with preinstalled InfiniBand drivers and Intel MPI library runtime
   --vm-sku Standard_H16r – InfiniBand-enabled VM instance H16r
   --instance-count 4 – number of VMs in the scaleset

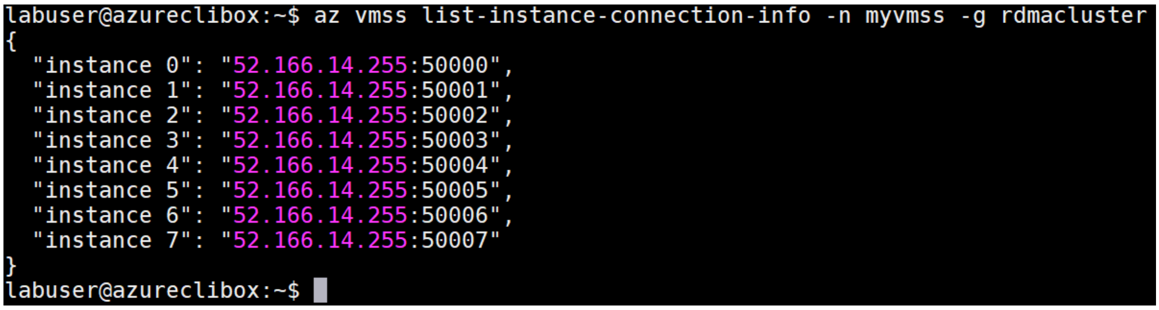
   It will take about 2 minutes to deploy the VM scaleset. Your public SSH key from *~/.ssh/id_rsa.pub* on the Azure CLI host will automatically be deployed to the scaleset as an authorized key for accessing the cluster nodes.
   On successful creation of the VM scaleset you should see output similar to the following:



```
labuser@azureclibox:~$ az vmss create -n myvmss -g rdmacluster --image OpenLogic:CentOS-HPC:7.1:latest
8
{
  "vmss": {
    "overprovision": true,
    "provisioningState": "Succeeded",
    "singlePlacementGroup": true,
    "uniqueId": "6dd433af-ad86-4385-9d48-a3b20061624f",
    "upgradePolicy": {
      "automaticOSUpgrade": false,
      "mode": "Manual"
    },
    "virtualMachineProfile": {
      "networkProfile": {
        "networkInterfaceConfigurations": [
          {
            "name": "myvms268dNic",
            "properties": {
              "dnsSettings": {
                "dnsServers": []
              },
              "enableAcceleratedNetworking": false,
              "enableIPForwarding": false,
              "ipConfigurations": [
                {
                  "name": "myvms268dIPConfig",
                  "properties": {
                    "loadBalancerBackendAddressPools": [
```

5. Display connection information for the VM scaleset:

```
$ az vmss list-instance-connection-info -n myvmss -g rdmacluster
```



```
labuser@azureclibox:~$ az vmss list-instance-connection-info -n myvmss -g rdmacluster
{
  "instance 0": "52.166.14.255:50000",
  "instance 1": "52.166.14.255:50001",
  "instance 2": "52.166.14.255:50002",
  "instance 3": "52.166.14.255:50003",
  "instance 4": "52.166.14.255:50004",
  "instance 5": "52.166.14.255:50005",
  "instance 6": "52.166.14.255:50006",
  "instance 7": "52.166.14.255:50007"
}
labuser@azureclibox:~$ 
```

Your scaleset cluster is deployed into a private network. Together with the scaleset Azure by default has created a load-balancer with public IP and NAT mapping for ports from 50000 upwards to subsequent nodes in the scaleset.

Note down public IP address of the load balancer which will be needed in the following steps to access the scaleset custer nodes.

6. Test access to your scaleset cluster by SSH login to any of the nodes:

```
$ ssh 52.166.14.255 -p 50000
$$ exit
$
```

Step 2: Configure SSH cluster connectivity

7. Run the below script *sshkey_copy.sh* on your Azure CLI box to distribute your private SSH key to the scaleset nodes:

sshkey_copy.sh

```
#/bin/bash
# Usage ./sshkey_copy.sh <publicip> <#nodes>

PIP=$1
COUNT=$2

if [ $# != 2 ]; then
    echo "Usage: $0 <publicip> <#nodes>"
    exit 1
fi
```

```
# compose SSH config file with strict host key checking turned off
echo 'Host *' > /tmp/config
echo 'StrictHostKeyChecking no' >> /tmp/config
chmod 400 /tmp/config

# copy private key and config to scaleset nodes
for (( i=50000; i<50000+$COUNT; i++ )); do
  scp -o StrictHostKeyChecking=no -P $i ~/.ssh/id_rsa $PIP:.ssh/
  scp -o StrictHostKeyChecking=no -P $i /tmp/config $PIP:.ssh/
done

rm -f /tmp/config
```

```
$ sshkey_copy.sh 52.166.14.255 8
```

Now your scaleset cluster nodes all share the same private key and can communicate with one another over SSH using passwordless key authentication.

Step 3: Test RDMA InfiniBand connectivity within the scaleset cluster

8. Login to any node in the scaleset, e.g. on port 50000:
   ```
   $ ssh 52.166.14.255 -p 50000
   ```

9. Install nmap package on the compute node:
   ```
   $$ sudo yum install -y nmap
   ```

10. Generate the list of cluster node IPs with the following command:
    ```
    $$ nmap -sn 10.0.0.* | grep 10.0.0. | awk '{print $5}' | tee
    nodeips.txt
    ```
    The list of node IPs is stored in file nodeips.txt in the home directory.

11. Use the below script *pingpong.sh* to run Intel MPI pingpong test between each pair of cluster nodes:

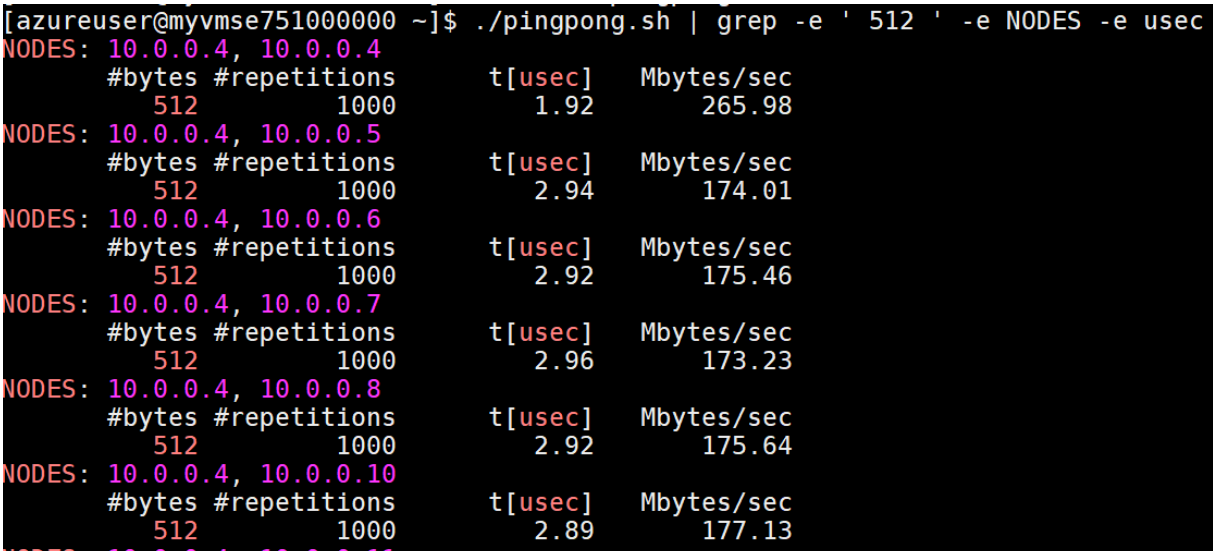
pingpong.sh

```
#!/bin/bash
# Example usage: ./pingpong.sh | grep -e ' 512 ' -e NODES -e usec

source /opt/intel/impi/`ls /opt/intel/impi`/bin64/mpivars.sh

for NODE in `cat ~/nodeips.txt`; do
    for NODE2 in `cat ~/nodeips.txt`; do
        echo '##################################################'
&& \
        echo NODES: $NODE, $NODE2 && \
        echo '##################################################'
        mpirun -hosts $NODE,$NODE2 -ppn 1 -n 2 \
            -env I_MPI_FABRICS=dapl \
            -env I_MPI_DAPL_PROVIDER=ofa-v2-ib0 \
            -env I_MPI_DYNAMIC_CONNECTION=0 \
            IMB-MPI1 pingpong
    done;
done
```

Run the following command:

```
$$ ./pingpong.sh | grep -e ' 512 ' -e NODES -e usec
```

You should expect the following output:



```
[azureuser@myvmse751000000 ~]$ ./pingpong.sh | grep -e ' 512 ' -e NODES -e usec
NODES: 10.0.0.4, 10.0.0.4
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         1.92       265.98
NODES: 10.0.0.4, 10.0.0.5
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         2.94       174.01
NODES: 10.0.0.4, 10.0.0.6
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         2.92       175.46
NODES: 10.0.0.4, 10.0.0.7
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         2.96       173.23
NODES: 10.0.0.4, 10.0.0.8
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         2.92       175.64
NODES: 10.0.0.4, 10.0.0.10
       #bytes #repetitions      t[usec]   Mbytes/sec
          512         1000         2.89       177.13
```

with latency measurements ~3usec proving RDMA InfiniBand connectivity between the nodes.

12. Exit the cluster node:

```
$$ exit
```

Step 5: Clean up resources

13. As the last step of the exercise clean up the cluster by deleting the resource group where it belongs:

```
$ az group delete -n rdmacluster
```